Package: prettycode (via r-universe)

July 1, 2024

Title Pretty Print R Code in the Terminal

Version 1.1.0.9000

Description Replace the standard print method for functions with one that performs syntax highlighting, using ANSI colors, if the terminal supports them.

License MIT + file LICENSE

URL https://github.com/r-lib/prettycode#readme, https://r-lib.github.io/prettycode/

BugReports https://github.com/r-lib/prettycode/issues

Depends R (>= 3.6)

Imports crayon, utils

Suggests covr, mockery, rstudioapi, testthat (>= 3.0.0), withr

Config/Needs/website tidyverse/tidytemplate

Config/testthat/edition 3

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Repository https://r-lib.r-universe.dev

RemoteUrl https://github.com/r-lib/prettycode

RemoteRef HEAD

RemoteSha 731d3fe332c39a8d544142067d59b253f74a2e85

Contents

1

---

default_style   *Default* prettycode *syntax highlighting style*

Description

A style function must return a named list of functions. Possible entries:

- reserved: reserved words
- number: numeric literals
- null: the NULL constant
- operator: operators, including assignment
- call: function calls
- string: character literals
- comment: comments
- bracket: brackets: (){}[]

Usage

```
default_style()
```

Details

Each entry in a list must be a function that takes a character scalar, and returns a character scalar with the exception of bracket which should be a list of these type of functions defining a color sequence. The default style adds ANSI formatting to the code.

Note that you can also change the code if you like, e.g. to include a unicode arrow character instead of the two-character assignment operator.

Examples

```
highlight(deparse(get), style = default_style())
```

---

highlight   *Syntax highlight R code*

Description

Syntax highlight R code

Usage

```
highlight(code, style = default_style())
```

Arguments

code	Character vector, each element is one line of code.
style	Style functions, see [default_style()]().

Value

Character vector, the highlighted code.

Examples

```
highlight(deparse(ls))
cat(highlight(deparse(ls)), sep = "\n")
```

---

prettycode	*Pretty Print R Code in the Terminal*

Description

Replace the standard print method for functions with one that performs syntax highlighting, using ANSI colors, if the terminal supports them.

Usage

```
prettycode(warn_conflicts = TRUE)
```

Arguments

warn_conflicts	logical. If TRUE, warnings are printed about conflicts from replacing the standard print method.

---

print.function	*Print a function with syntax highlighting*

Description

To turn on pretty printing of functions, you need to call prettycode::prettycode(). It might be a good idea to call it from your .Rprofile.

Usage

```
## S3 method for class '`function`'
print(x, useSource = TRUE, style = default_style(), ...)
```

Arguments

x	Function to print.
useSource	Whether to use the stored source code, if available.
style	The highlight style to use, see default_style().
...	Not used currently, for compatibility with the print generic.

Value

The function, invisibly.

Index